AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions and listings of claims in the application:

1.-35. (Canceled)

36. (New) A computer-implemented method, performed by a computer system comprising one or more processors and computer memory, for modifying information in a database stored in volatile memory, comprising:

receiving a modification request including a search string and modification data;

assigning the modification request to an update thread for processing, the

processing comprising:

using a search engine, retrieving a first pointer to a first record

corresponding to the search string;

using the first pointer, copying the record into a second record;

modifying the data in the second record based on the modification data;

determining a second pointer for the second record; and

in an uninterruptable process and without locking read access to the

database, updating the search engine with the second pointer;

writing the second record to a snapshot file stored in non-volatile memory; and

purging the first record from the database stored in volatile memory, the

snapshot file maintaining the first and second record.

37. The method of claim 36, wherein writing occurs after at least two different modification requests update the pointer contained in the search engine.

38. The method of claim 36, wherein writing occurs when the database stored in volatile memory grows to a limit allocated for the database.

39. The method of claim 36, wherein the search engine comprises a hash table.

40. The method of claim 39, wherein the hash table is rebuilt when the purging occurs.

41. The method of claim 36, wherein the snapshot file is loaded from a remote computer across a network.

42. A system for modifying information in a database stored in volatile memory comprising:

a non-transitory memory storing instructions; and

a processor executing the instructions to cause the system to perform a method comprising:

receiving a modification request including a search string and modification data;

assigning the modification request to an update thread for processing, the processing comprising:

using a search engine, retrieving a first pointer to a first record

corresponding to the search string;

using the first pointer, copying the record into a second record;

modifying the data in the second record based on the modification

data;

determining a second pointer for the second record; and

in an uninterruptable process and without locking read access to

the database, updating the search engine with the second

pointer;

writing the second record to a snapshot file stored in non-volatile memory;

and

purging the first record from the database stored in volatile memory, the

snapshot file maintaining the first and second record.

43.   The system of claim 42, wherein writing occurs after at least two different

modification requests update the pointer contained in the search engine.

44.   The system of claim 42, wherein writing occurs when the database stored in

volatile memory grows to a limit allocated for the database.

45.   The system of claim 42, wherein the search engine comprises a hash table.

46.   The system of claim 45, wherein the hash table is rebuilt when the purging

occurs.

47.   The system of claim 42, wherein the snapshot file is loaded from a remote

computer across a network.

48.   A non-transitory computer-readable storage medium containing instructions

which, when executed on a processor, perform a method comprising:

        receiving a modification request including a search string and modification data;

        assigning the modification request to an update thread for processing, the

                processing comprising:

                using a search engine, retrieving a first pointer to a first record

                        corresponding to the search string;

                using the first pointer, copying the record into a second record;

                modifying the data in the second record based on the modification data;

                determining a second pointer for the second record; and

                in an uninterruptable process and without locking read access to the

                        database, updating the search engine with the second pointer;

        writing the second record to a snapshot file stored in non-volatile memory; and

        purging the first record from the database stored in volatile memory, the

                snapshot file maintaining the first and second record.

49.   The non-transitory computer-readable storage medium of claim 48, wherein

writing occurs after at least two different modification requests update the pointer

contained in the search engine.

50.   The non-transitory computer-readable storage medium of claim 48, wherein

writing occurs when the database stored in volatile memory grows to a limit allocated for

the database.

51.   The non-transitory computer-readable storage medium of claim 48, wherein the

search engine comprises a hash table.

52.   The non-transitory computer-readable storage medium of claim 51, wherein the

hash table is rebuilt when the purging occurs.

53.   The non-transitory computer-readable storage medium of claim 48, wherein the

snapshot file is loaded from a remote computer across a network.